

Lab Assignment



Cybersecurity Professional Program
Introduction to Python
for Security

File System & Error Handling

PY-04-L5
**OS Module &
Open Function**



Lab Objective

Understand how to interact with the operating system and perform file operations in Python.



Lab Mission

Save a *ping* output to a file with Python's **OS** module.



Lab Duration

20–40 minutes



Requirements

- Basic knowledge of the **OS** module
- Familiarity with the *open()* function



Resources

- Environment & Tools
 - Windows
 - PyCharm
 - Python 3



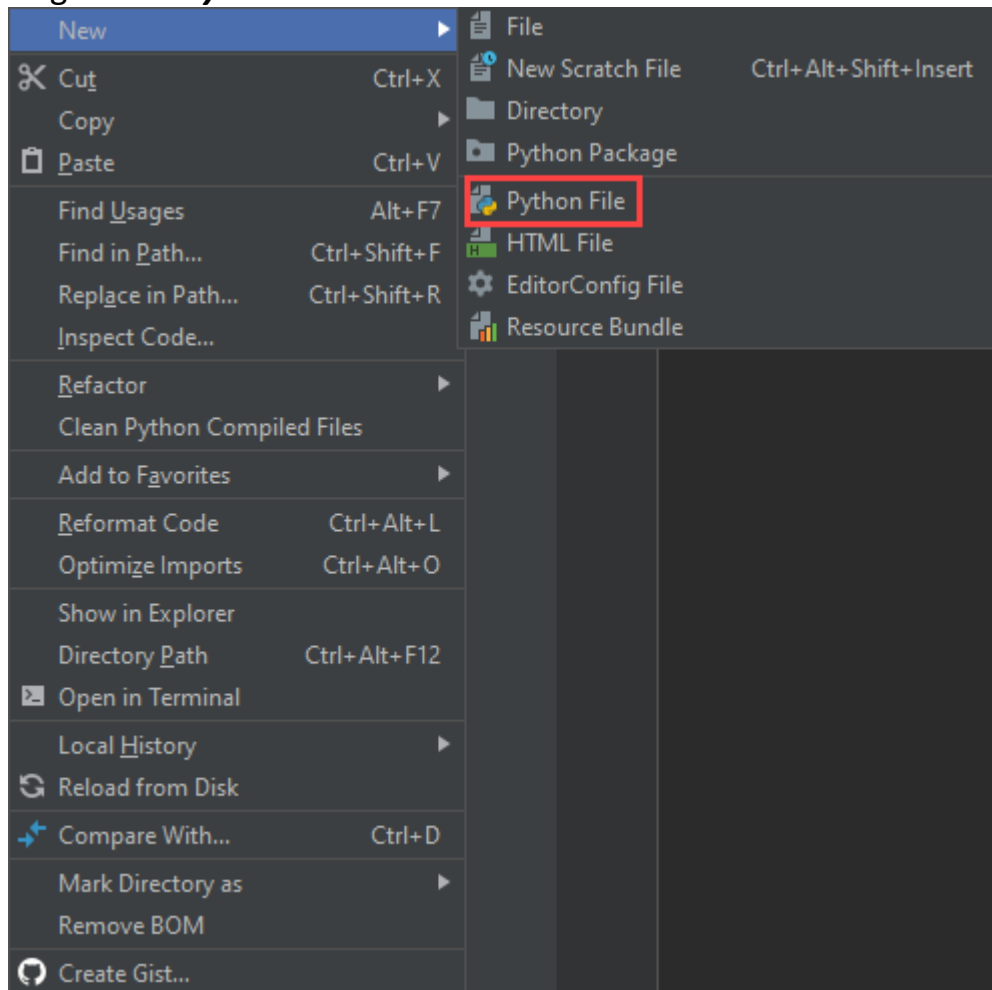
Textbook References

- Chapter 4: File System and Error Handling
 - Section 3: Module Definition and Usage

Lab Task: OS Module & open() Function

In this task, you will perform a ping to **8.8.8.8** to verify that a network connection exists in the system.

- 1 Create a new Python file in PyCharm by right-clicking the project you created and selecting **New > Python File**.



- 2 Create a variable and assign it a **filename** value provided by the user.

```
file_name = input("Choose a filename: ")
```

3 Import the OS library.

```
import os

file_name = input("Choose a filename: ")
```

4 Use the **os.system()** function to ping the public IP address **8.8.8.8**.

```
import os

file_name = input("Choose a filename: ")
#for mac and linux add -c 4 flag
os.system(r"ping 8.8.8.8")
```

5 Save the results to a new file with the name chosen by the user.

Note: Since the **os.system()** function executes commands on the operating system, you need to rely on **stdout** operations, such as output redirects, to create and append to a file using the symbol **>>**. In addition, remember that internet connectivity can be validated by pinging a public IP address.

```
import os

file_name = input("Choose a filename: ")
#for mac and linux add -c 4 flag
os.system(r'ping 8.8.8.8 >> "' + file_name + '".txt')
```

6 Use the **open()** function to read the file that was created in the previous step.

```
import os

file_name = input("Choose a filename: ")
#for mac and linux add -c 4 flag
os.system(r'ping 8.8.8.8 >> "' + file_name + '".txt')
with open(file_name + ".txt", "r") as file:
```

7 Create an *if* condition to check if the file contains “ms.”

```
import os

file_name = input("Choose a filename: ")
#for mac and linux add -c 4 flag
os.system(r'ping 8.8.8.8 >> ' + file_name + '.txt')
with open(file_name + ".txt","r") as file:
    if "ms" in file.read():
```

8 Print a message to inform the user that internet connectivity is available.

Note: Observe the output of the *ping* command when an IP address is reachable, as opposed to when it is unreachable. Each outcome will output a string that can be used in the code to create a condition statement.

```
import os

file_name = input("Choose a filename: ")
#for mac and linux add -c 4 flag
os.system(r'ping 8.8.8.8 >> ' + file_name + '.txt')
with open(file_name + ".txt","r") as file:
    if "ms" in file.read():
        print("You have an internet connection")
```

9 Add an *else* statement to print a message if there is no internet connection.

```
import os

file_name = input("Choose a filename: ")
#for mac and linux add -c 4 flag
os.system(r'ping 8.8.8.8 >> ' + file_name + '.txt')
with open(file_name + ".txt","r") as file:
    if "ms" in file.read():
        print("You have an internet connection")
    else:
        print("You don't have an internet connection")
```

10 Open the file and view the result of the ping.